NRL Report 8155

Security/Privacy Evaluation Subcommittee Report on the Candidate Message Service Systems for the Military Message Experiment

S. H. Wilson
Naval Research Laboratory

S. R. Ames, Jr., and J. D. Tangney
The MITRE Corporation
Bedford, Massachusetts

and

J. R. Bunch, Jr.
CINCPAC Staff
Camp Smith, Oahu, Hawaii

September 14, 1977

NAVAL RESEARCH LABORATORY
Washington, D.C.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Report 8155	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) SECURITY/PRIVACY EVALUATION SUBCOMMITTEE REPORT ON THE CANDIDATE MESSAGE-SERVICE SYSTEMS FOR THE MILITARY MESSAGE EXPERIMENT		5. TYPE OF REPORT & PERIOD COVERED Interim Report on continuing NRL Problem 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) S. H. Wilson, S. R. Ames, J. D. Tangney, and Joseph Bunch		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory, Wash., D.C. 20375 MITRE Corp., Bedford, Massachusetts 01730 Commander in Chief, Pacific		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NRL Problem B02-40 64510N, X-0743-CC
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Electronic Systems Command Washington, D.C. 20360		12. REPORT DATE September 14, 1977 13. NUMBER OF PAGES 42
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Security, Kernel, MME, DISTAN, Messages, Evaluation, Certification		

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The military message-handling experiment (MME) is a part of a research and development program within the Navy and the Defense Advanced Research Projects Agency (DARPA) whose goal is the development of advanced message-handling systems for the military. The purpose of the MME is to evaluate the use of a secure computer-aided message service at CINCPAC Headquarters, Camp Smith, Oahu, Hawaii. During 22 February through 3 March 1977, candidate message services for the experiment were submitted by Bolt, Beranek, and Newman, Inc. (BBN), the Massachusetts Institute of

(Continued)

20. (Continued)

Technology (MIT), and the Information Sciences Institute (ISI) of the University of Southern California and were evaluated by representatives from the Navy, DARPA, MITRE, CTEC, Inc., and the CINCPAC staff.

Because an automated military message service must process messages of multiple levels of classification, issues of security and privacy exerted a major influence in the evaluation and selection process. The results of the security/privacy evaluations of the candidate services are documented in this report of the Security/Privacy Evaluation Subcommittee.

The Security/Privacy Evaluation Subcommittee recommends that ISI's Sigma message service be selected for use in the MME. This recommendation is based on three major considerations: user interface, secure design, and appropriateness for future secure message services.

CONTENTS

SECURITY/PRIVACY EVALUATION SUBCOMMITTEE REPORT ON THE CANDIDATE MESSAGE-SERVICE SYSTEMS FOR THE MILITARY MESSAGE EXPERIMENT

1. INTRODUCTION

The military message-handling experiment (MME) is a part of a research-and-development program within the Navy and the Defense Advanced Research Projects Agency (DARPA) whose goal is the development of advanced message-handling systems for the military. The military message-handling experiment is an attempt to evaluate the use of a computer-aided message service in an operational environment. The terms of reference for the experiment are contained in a Memorandum of Agreement [1] between the DARPA, the Naval Electronic Systems Command (NAVELEX), the Naval Telecommunications Command (NAVTELCOM), and the Commander in Chief, Pacific (CINCPAC).

Candidate message service systems for the experiment were submitted by Bolt, Beranek, and Newman, Inc. (BBN), the Massachusetts Institute of Technology (MIT), and the Information Sciences Institute (ISI) of the University of Southern California. During 22 February through 3 March 1977, representatives from the Navy, DARPA, MITRE, CTEC, Inc., and the CINCPAC staff evaluated the three candidate message services for installation at CINCPAC Headquarters, Camp Smith, Oahu, Hawaii.

Because an automated military message service must process messages of multiple levels of classification, issues of security and privacy exerted a major influence in the evaluation and selection process. The results of the security/privacy evaluations of the candidate services are documented in this report of the Security/Privacy Evaluation Subcommittee.

Background material on the MME, computer system security, and specific security considerations of the MME is presented in Section 2. Brief overviews of the security/privacy evaluation and of the three candidate message services are in Section 3. Results of the security/ privacy evaluation are documented in Sections 4, 5, and 6. The subcommittee's conclusions are in Section 7. The scoring is summarized in Appendix A.

2. BACKGROUND

2.1 Military Message Experiment

The immediate goal of the military message-handling experiment is to develop a military message system for handling formal and informal message traffic on an experimental basis in a military environment. The longer range goal is to use the experimental system to resolve complex questions concerning the military use of such a system, the user interface, the ramifications of security

Manuscript submitted June 30, 1977.

issues, and hardware and software architectures. The insights gained in this experiment will be used in designing an operational message-processing system for the Department of Defense.

The candidate systems submitted were implemented on a TENEX and used Hewlett-Packard alphanumeric displays. Because the developers were not required to implement their systems in a secure manner for the experiment, each developer submitted a security design to indicate how the system could be implemented in a secure manner preserving the user interface of the TENEX version. The subcommittee evaluated each of these message systems and its security design as a candidate for a verifiably secure message-processing system operating in a multilevel security environment.

The site of the experiment is CINCPAC Headquarters, Camp Smith, Hawaii. Presently at CINCPAC, a manual message system supports the distribution, filing, preparation, coordination, and release of formal messages for transmission on the AUTODIN network. Messages are received from and transmitted to the AUTODIN network via an AUTODIN Local Digital Message Exchange (LDMX) terminal. Except for AUTODIN and the LDMX, the CINCPAC system is largely manual; messages are typed on paper, distributed by runners, and filed in folders stored in file cabinets and safes. A computer-based system will put the entire service on line; users will access the service by CRT terminals, and messages will be distributed, filed, prepared, coordinated, and released on line by user commands to the computer system. The interactive message service will support informal message traffic among CINCPAC users, as well as formal AUTODIN message traffic.

A description of current manual message-handling procedures at CINCPAC is contained in Ref. 2, and a discussion of the functions that an interactive service should provide is contained in Ref. 3. CINCPAC users of the current system will be trained to use the interactive message service, and various tests will be conducted to determine the utility of the interactive service and the overall impact of the service on CINCPAC operations. Operational tests of both the manual and interactive systems will determine the usefulness of the computer-based system. Structural tests will identify desirable capabilities of an interactive system and indicate useful features at the user interface. Organizational impact tests will evaluate user acceptance of the interactive system. The overall MME test plan is documented in Ref. 4, and the specific test procedures in Ref. 5.

2.2 Security/Privacy

Formal AUTODIN message traffic consists of military messages at classification levels ranging from Unclassified through Top Secret. To serve the operational needs of the CINCPAC community effectively, the interactive message service must maintain the integrity of this multilevel information. At a minimum, users must be confident that the privacy of message traffic is protected and that messages are not accessible to other users until proper coordination and release have been accomplished. Further, to prevent highly sensitive information from being compromised, particularly if the service is accessible to users of differing levels of clearance, effective information-access control must be enforced on the basis of user clearance, information classification, and need to know.

Many approaches to the problems of providing a secure computer facility have been proposed. For a survey, see Ref. 6. There does not yet exist any proven solution. Some concepts are intuitively obvious. First, a consistent implementable security policy must be articulated. This policy must be translated into computer program specifications and then into programs. That portion of the software that is security relevant (i.e., a deviation in the performance from the specification of this software could allow a security violation) must be identified. This security-relevant software should be verified. In general, the total verification of the software is not possible, because there is too much security-relevant code. A compromise approach is to verify that portion that controls the access to and passing of information and to monitor the actions of other security-relevant software. The approach taken in the MME has been to adopt the security kernel philosophy towards security. The kernel approach is described in Refs. 7, 8, and 9. The main thrust of research into security kernels was sponsored by the Air Force Electronic Systems Division (ESD).

2.2.1 Security-Kernel Concept

Many of the results of the ESD work are being used in the MME. Specifically, each of the three message services is being designed to follow the rules of a mathematical model used to describe the hierarchical relationships of the four security classifications (Unclassified through Top Secret) of the DOD. The model is based on the concept of a reference monitor -- an abstract mechanism that controls the flow of information within a computer system by mediating every attempt by a subject (active system element) to access an object (information). (In a computer system, subjects are users and processes, and objects include programs, data files, and peripheral devices.) The hardware-software mechanism that implements the reference monitor is called a security kernel. The security kernel uses the rules of the mathematical model as a specific policy in mediating access requests. This incorporation of policy into the kernel allows for the possibility of a proof that the kernel correctly applies the policy to the information it protects.

2.2.2 Model Axioms

The mathematical model [7,8] establishes an "inductive nature" of security by demonstrating the preservation of security from one system state to another. The model defines security with two axioms: the simple security condition and the *-property [10]. The simple security condition states that a subject cannot observe an object unless the security level of the subject is greater than or equal to the security level of the object. (A security level is composed of a classification and a set of compartments. One security level is considered greater than or equal to a second security level if (a) the classification of the first is greater than or equal to the classification of the second and (b) the set of compartments of the first is a superset of the set of compartments of the second.) The intent of the simple security condition is obvious: to prohibit users from obtaining information that they are not cleared to see.

The *-property further restricts possible access by stipulatin that a subject may not modify an object if that object has a security level lower than the security level of the subject. The *-property is designed to prohibit a program that is operating on behalf of a user from reducing the classification of any information. When a user is given a clearance, he is charged with responsibility for maintaining the classification of classified information. Normally, when the user is working with pencil and paper, we trust the tools that he is working with not to compromis information. Obviously, the user has a strong and direct control over the pencil and paper. However, the tools that a computer utility may provide cannot be similarly trusted. This is due to the very limited and indirect control that the user has over the software operating on his behalf, the amount of information that may be compromised, the speed with which the compromise may occur, and the diffculty in detecting the violating program. By enforcing the *-property on computer software, a program will not be able to either accidentally or maliciously compromise information. (Designers of computer utilities constrained by the *-property must ensure that *-property enforcement does not unnecessarily restrict the capabilities of the user.)

The "inductive nature" of security established by the model may be defined as follows: if the system can be shown to exist in an initial secure state and if all system primitives are defined in accordance with model axioms, then all subsequent system states -- effected by execution of system primitives -- can also be shown to be secure.

2.2.3 Kernel Requirements

To provide security, a kernel must mediate every access by a subject to an object, be protected from unauthorized modification, and correctly perform its functions. A kernel satisfies the first requirement by creating an environment in which all nonkernel software is constrained to operate and by maintaining control over this environment. The kernel creates an abstract or virtual machine whose operations include the basic hardware machine instructions and primitives implemented by kernel software. Since the kernel primitives provide the only means of accessing the objects of the system (in accordance with the model axioms), all nonkernel software is constrained to operate through the kernel, insuring the mediation of all subject-object accesses.

If a security policy is correctly built into the abstract machine, then programs running on it will not be able to perform operations that violate the policy. In practice, the abstract machine created by a security kernel will include all of the unprivileged machine instructions of the base hardware, constrained by a hardware-supported memory protection mechanism.

The requirement for protection against unauthorized modification is satisfied by isolating security kernel software in one or more protection domains. As an example, a ring mechanism [11] can be used to provide a domain protected from unauthorized modification.

Finally, the requirement that the kernel correctly performs its functions is satisfied by using a formal methodology to demonstrate its correctness. A suitable methodology was introduced by Bell and Burke [12]. Basically, there are two steps: a proof that kernel behavior enforces the desired security policy and a proof that the kernel is correctly implemented with respect to the description of its behavior used in the first step. Kernel behavior can be described with a nonprocedural program specification. A method for proving that a kernel specification is a valid interpretation of a mathematical model of a security policy has been developed by Ames [13] and Millen [14]. Techniques developed by a group at the Stanford Research Institute (SRI) can be used to prove that the implementation of a kernel (or any other computer program) is correct with respect to its specification [15, 16]. The SRI techniques involve the decomposition of the kernel into hierarchically structured levels of abstraction.

2.3 MME Security Goals

There are two primary security goals for the MME: the development of a usable man/machine interface to a multilevel message service and the identification of necessary security kernel primitives for a secure multilevel message service.

A security kernel has not been developed to secure the TENEX operating system of the PDP-10 host machine for the MME. Rather, MITRE and BBN designed a series of security enhancements to TENEX called AIM (Access Isolation Mechanism). The purpose of AIM was not to provide a verifiably secure file system for TENEX but rather to enhance the security controls. Additionally, AIM-enhanced TENEX would simulate a kernel-based secure TENEX, permitting a thorough analysis and evaluation of the design and implementation of a secure multilevel message service on a secure operating system.

However, based on the directions some of the message-service developers were pursuing, it became clear that the object size protected by AIM could constrain their development of a secure message service. Consequently, the developers were not required to use the AIM enhancements; instead, they were given the latitude of incorporating the fundamental access controls within the software of the message-service application, and all of the developers chose to do so.

2.3.1 Usable Security Interface

Each developer delivered for evaluation a message service whose user interface reflected the access controls of a secure multilevel message service. Although the access controls were implemented within message-service code and, hence, could not be formally verified as correct or inviolable, the MME will take place within a strictly controlled Top Secret environment where all users are cleared to Top Secret and all information processed by the message service is protected at Top Secret. (As a result of the protection of all messages at Top Secret, the release of messages to AUTODIN is a security-sensitive issue, and additional controls are necessary to verify that the classifications affixed to released messages, ranging from Unclassified to Top Secret, reflect the true sensitivity of the information contained within the message.) Thus, we presume that the integrity of the access controls will not be challenged by an untrustworthy individual. Furthermore, each outgoing message will be manually reviewed.

The experiment offers the opportunity to study user interaction with a model of a secure military message processing system. The purpose of the security/privacy evaluation guidelines and the actual evaluation were to encourage the developers to produce a system that integrates a powerful message processor with the required security controls and a pleasing and effective user interface. A system of this type would encourage the use of the system and would maximize the information that can be gained from observing the user/system interactions. Some of the expected results from the experiment are specifications for the user interface and a better understanding of the hardware and software architectures needed to produce a secure message-processing system for military applications.

2.3.2 Security Kernel Primitives

The other MME security goal is the identification of the primitive functions that a security kernel must include to support a secure message service for the multilevel user environment. A multilevel user environment may range from a controlled two-level environment (e.g., Secret and Top Secret users) where some personnel and administrative controls are exercised to the open environment where uncleared users may access the service concurrently with cleared users and where no personnel or administrative controls are exercised on the uncleared users.

To address this second goal, each developer was required to deliver for evaluation a design for a secure message service, predicated on the existence of a secure operating system based on a verified security kernel. A large measure of the security/ privacy evaluation is directed toward an examination of each design's demands on the functional capabilities of a security kernel necessary to support an implementation of a secure multilevel message service in a multilevel user environment.

It was further required that the design be able to support the user interface presented by the developer's candidate message service. This requirement for compatibility links the two security goals together. Decisions that the developer makes in designing the secure message service have a direct bearing on the types of security features provided at the user interface, and the features have a direct bearing on the usability of the security interface.

2.4 Features of a Secure Military Message Service

The remainder of this section is devoted to a short description of the features that an interactive secure military message service should provide. The concept of a multilevel display terminal is also introduced.

2.4.1 Components of a Message

The types of features that a secure military message service must support include the ability to interactively read, create, file, retrieve, and annotate messages at various security levels. With approximately 1000 messages being received a day, selective retrieval of a message on the basis of keyword, date-time group, originator, and subject is required. Given the necessary features that must be supported by a military message service, a formal military message must be viewed as a multilevel object with an overall message classification and various components, or fields, at possibly different classifications equal to and below the message classification. The components of a military message are as follows:

- Header: The header of a message contains the address, destination, orginator, date-time-group identifier, overall message classification marking, etc. The content of a message header is defined to be unclassified.

- Subject: The subject of a message may be of any classification less then or equal to the overall message classification.

- References: Most references are unclassified and contain a date-time-group and originator. However, certain references may contain additional information that, unless specifically classified, must be treated at the overall classification of the message.

- Text: The text of the message has a classification less than or equal (but usually equal) to the overall message classification. Paragraphs of the text may be labeled at classifications equal to or lower than the overall classification of the text.

- Annotations: Although not specifically part of a message, annotations at any security classification may be added by a user to a message or specific message field.

- Keywords: For filing and retrieval purposes, keywords can be added by the user to the message. Keywords may be of any classification.

2.4.2 Text Objects and Message Selectors

Classified text objects and message selectors are desirable features. Text objects are simply character strings, such as address lists and commonly used pieces of text. There should be some facility to excise and save text from an existing message as a text object for later uses in message preparation. Reclassification of text objects is a convenient feature, provided all reclassification operations are fully audited.

Message selectors are user-defined message selection criteria that are useful for common message search and retrieval operations. It should be possible to define permanent selectors (e.g., as part of the user profile) that contain often-used subjects, keywords, etc., in search operations.

2.4.3 Multilevel Terminal

To work effectively, the user of the military message service must be able to easily read and write messages at different levels of classification. For instance, the user may wish to refer to an existing Top Secret message while composing an Unclassified reply. To facilitate such an operation, and to satisfy the more general requirement for a usable security interface, the MME has adopted the concept of a multilevel display terminal.

The multilevel terminal is implemented on an Intel-8080-microprocessor-based Hewlett-Packard 2649, consisting of 48K bytes of program memory (RAM, ROM, or PROM) and 12K bytes of RAM display memory. ISI has developed a multilevel terminal program that effectively partitions display memory into as many as seven distinct windows. Each window can be independently classified and displayed on the terminal screen under the control of the message service on TENEX. Using the instance above, the message service would allocate a window at Top Secret, write the Top Secret message into it, and display it on the screen; then, the service would allocate a window at Unclassified, write a message composition form into it, and display it on the screen. The message service determines the windows to be displayed and their locations on the display.

The multilevel terminal raises a number of additional security concerns: the effective marking of multilevel information being read or written; security controls within the terminal program to protect against information compromise within the terminal, and controls to ensure a secure transmission path between the terminal and message-service processes at the appropriate security levels. These issues are addressed further in Section 5.

Both MIT and ISI chose to use the multilevel terminal as the basis for their security user interfaces. MIT requested several terminal program modifications that were designed to tailor the terminal to some specifics of the MIT message service. BBN opted not to use the multilevel terminal; instead they used a Hewlett-Packard 2645 modified to transmit additional function keys.

3. OVERVIEWS

This section contains a brief overview of the security/privacy evaluation and brief overviews of the message services that were evaluated.

The evaluation of the three candidate services began on 22 February and continued until 1 March. The plan was to evaluate the three services in sequence - BBN, ISI, and MIT - allocating a three-day period for each service. The BBN and ISI services were fully evaluated, and the results of the security/privacy evaluation are documented in Sections 4, 5, and 6. MIT's MSG-DMS service was not evaluated. Evaluation of a partially implemented MSG-DMS was initiated, but this service did not perform well enough to permit its fair and complete evaluation. Hence, no evaluation of the MSG-DMS system or the MIT design of a secure service is documented in this report.

Five evaluation subcommittees were formed to evaluate each of the candidate services. The five subcommittees were:

- functional requirements,
- human factors,
- performance,
- training/documentation, and
- security/privacy.

Each of the five subcommittees conducted its own independent evaluation according to predetermined criteria and reported its results to the selection committee. At the conclusion of the evaluations, the selection committee met in Washington, D.C., on 14-16 March 1977, to review the results and announce the selected service for the MME at CINCPAC.

Members of the security/privacy evaluation subcommittee were:

- Stan Wilson, Naval Research Laboratory (Chairman),
- Stan Ames, MITRE,
- John Tangney, MITRE,
- Joe Bunch, CINCPAC Staff.

3.1 Security/Privacy Evaluation Criteria

A more complete treatment of the security/privacy evaluation criteria used during the evaluation is available in Ref. 17. Appropriate excerpts from Ref. 17 are included in this report to avoid the need for repetitive referencing.

Security/privacy evaluation criteria are divided into three areas; they are listed below along with the percentage each area contributes to the overall security/privacy score.

- MME security selection criteria (50%),
- Secure system structure (40%), and
- Certifiability (10%).

A total of 100 points are available in each area of evaluation. Specific criteria and the number of points each carries are documented in Sections 4, 5, and 6. The following procedure was used to determine the score for each system. Raw points scored in each area were multiplied by the area's percentage to determine a net score for the area. Net scores from the three areas were then added together to determine an overall security/privacy score, which was then mapped into a multiplier by the selection committee for its use in the selection process.

Each of the three areas of the security/privacy evaluation are summarized below.

3.1.1 MME Security Selection Criteria (50%)

A set of MME selection checks [3] forms the basis for the MME evaluation and selection process. The MME selection criteria document includes a section specifying the minimum set of security/privacy capabilities that a message service must support for operational use at CINCPAC. This evaluation determined both the degree to which each candidate message service satisfied the identified security/privacy selection requirements and the manner in which the requirements were supported at the user interface. The results of this area of evaluation are presented in Section 4.

3.1.2 Secure System Structure (40%).

The secure-system-structure area of evaluation dealt only with the designs for a secure message service and addressed the second goal of identifying the primitive functions that a kernel must provide to support a secure service. Four functional kernel areas were identified - secure file system, secure process structure, secure multilevel terminal design/multiplexer, and secure process coordination. The demands of each design on the required capabilities in the four areas were evaluated, and the compatibility of the design with the user interface presented by the developer's candidate service was considered. The results of this area of evaluation are documented in Section 5.

3.1.3 Certifiability (10%)

The third area of evaluation considered the problems in certifying an eventual implementation of each developer's design. During certification, the threats to system security posed by a particular user environment are weighed against the effectiveness of the security measures. In this evaluation, the subcommittee was interested in the certifiability of an eventual implementation that would permit users of various clearances to concurrently process messages in a multilevel mode. The certifiability of an implementation in four user environments was considered: strictly controlled, controlled, semicontrolled, and open. The results of this area of evaluation are discussed in Section 6.

(The concurrent processing of messages in a multilevel mode by users of various clearances is an operation under a supervisor or executive program which permits various levels and categories or compartments of material to be concurrently stored and processed in an ADP system. In a remotely accessed resource-sharing system, the material can be selectively accessed and manipulated from variously controlled terminals by personnel having different security clearances and access approvals. This mode of operation can accommodate the concurrent processing and storage of two or more levels of classified data or of one or more levels of classified data with unclassified data, depending on the constraints placed on the systems by the Designated Approving Authority.)

3.2 The Candidate Services

The remainder of this section contains a brief description of BBN's Hermes message service and ISI's Sigma message service. The services are considered and compared in terms of security interaction, message and message file protection, and terminal design.

3.2.1 BBN's Hermes Message Service

Interaction with Security

Of the two candidate services, BBN's Hermes system demands more user interaction with and awareness of security controls. With Hermes, the user must know his current operating security level. For example, the user logged on at Secret may operate at Unclassified, Confidential, and Secret. At the different operating levels, the user perceives no difference as far as interacting with Hermes (i.e., command interaction); rather, he perceives only a difference in the information that is accessible. When operatin at Confidential, only Unclassified and Confidential information can be read and only Confidential information can be written or edited. Typically, then, the user tends to operate at his log-on level (usually his highest level) for message reading and retrieval, switching to lower levels to create or edit messages. Additionally, the user profile (life-style switches, directories of accessible message-service objects) is maintained at Unclassified, so the user must switch to Unclassified to manipulate it.

Messages and Message Files

In Hermes, messages are protected on the basis of individual message fields, i.e., protection is enforced on each field of a message (e.g., subject, reference, annotation, keywords text). A message, then, is simply a collection of independently classified message fields. Although Hermes supports a message classification field, its contents are defined (and can be edited) by the user, and Hermes does not use the classification field in any way to control access to the message as a whole.

Message creation, therefore, is the process of creating and editing the various fields of the message and involves switching the Hermes service to the security level appropriate to the fields being composed.

With the Hermes approach to message protection, message fields are loosely organized into messages and stored in message files. Message files may contain fields at any classification, as there is no concept of message-file classification. To a user operating at Confidential, only the Unclassified and Confidential fields stored in an open message file are accessible. A security problem with this approach concerns the filing or moving of messages between message files. Consider a user, logged on at

Secret but operating at Confidential on message file ABC, wishing to file a copy of message 2 into another file, DEF. Message 2 consists of Unclassified header fields and a Confidential subject, all of which the user can see on a survey of ABC at Confidential, as well as a Secret text field, which the user cannot now see (but could if operating at Secret), and a Top Secret keyword, totally inaccessible to the user. (Message 2 could have been filed in ABC by some other user, operating at Top Secret during a previous session, who shares ABC with the current user.) Should the text and keyword be filed into DEF? Hermes does just that, and on a move command it proceeds to delete all of message 2 from ABC, including the text and keyword. Should nothing be filed, or should just the Unclassified and Confidential fields be filed? Enforcement of both a message and message file classification are clearly required in order to structure an acceptable solution.

Terminal design

Hermes does not use the ISI-developed multilevel terminal but instead uses a Hewlett-Packard 2645 that was modified to transmit the numeric function keys directly to TENEX. The terminal may operate in a multilevel mode, however, since information at various security levels up to the user's log-on level may be concurrently displayed.

Each line sent to the terminal for display is prefixed by an indicator character in column 1 which denotes the user's current operating security level. Transistions from one level to another are marked by placing the indicator character of the new level within an inverse video display token. Classifications of individual message fields being displayed are marked at the beginning of the field by an indicator character enclosed within square brackets.

In the upper right-hand corner of the terminal screen is a display that shows the maximum possible classification of any information currently on display. For example, a user operates at Secret for a while then switches to Unclassified. The upper displaywill indicate Secret until the user has worked long enough at Unclassified so that the last line prefixed by a Secret indicator character has rolled off the top of the screen. At this point, if the user is still working at Unclassified, the upper display will indicate Unclassified. In general, then, the upper display reflects the highest classified indicator character currently on display. The indicator characters denote only that information at the indicated level <u>may</u> be displayed on that line, and that the upper display denotes <u>only</u> that information at the indicated level <u>may</u> be displayed somewhere on the screen.

3.2.2 ISI's Sigma Message Service

Interaction With Security

Compared to the Hermes system, user interaction with the security aspects of ISI's Sigma message service are less overt. As there is no concept of current operating security level in Sigma, the user is not required to switch Sigma to predetermined operating levels in order to accomplish certain tasks. Rather, most commands and accompanying parameters are defined to be Unclassified. Typically, the user issues commands to an Unclassified command processor, and most of the user state (processing context) is maintained at Unclassified. All commands that must be processed at a classified level are implemented in terms of function keys that automatically switch Sigma's command processor and internal execution state to the appropriate level. At the completion of such a command, Sigma returns to its normal Unclassified mode.

Users specify a clearance at log-on, and for the duration of the session are permitted to access only messages, message files, and text objects that are classified at or below their log-on clearance.

Capabilities supported by the multilevel terminal provide the latitude and flexibility that permit Sigma's facile interface to security. With separately classifiable windows at the disposal of the message service, in a sense each window serves as a separate virtual terminal. Hence, Sigma can provide an Unclassified comman window for command entry, and the effects of these commands (e.g., print message, create message) take place within other, perhaps classified, windows.

Messages and Message Files

In Sigma, the message as a whole is the information object to which access is controlled. The entire contents of a message are protected at the level of the highest classified information contained in the message. For example, any Unclassified fields of a message that contains Secret text are effectively protected at Secret. If the user were to excise the Unclassified fields from the Secret message, Sigma would protect them as Secret objects. The user could then downgrade these objects to Unclassified for subsequent use.

Messages are composed in Sigma by issuing a create-message command, including a parameter that specifies the classification of the message. Sigma creates a terminal window at that classification and displays within it a form that includes the various fields of the type of message being composed (message type is also a parameter). The user simply fills in the desired fields, including classification markings on fields that are actually classified below the overall message classification. Sigma does not prompt the user whether to include these markings.

Message files consist of collections of citations (pointers) to messages within a common message data base shared by all users. Citations are simply handles on messages and are the items involved in message filing, moving, and forwarding operations.

Classifications are affixed to Sigma message files and access to message files is governed by nondiscretionary security policy. A Secret message file may contain only citations to Secret, Confidential, and Unclassified messages. Therefore, the user logged on at Unclassified can access neither the Secret message file nor any Unclassified messages it may contain.

Terminal Design

As noted, Sigma takes full advantage of the ISI-developed multilevel terminal. Indeed, Sigma's interface to security is as attractive as it is largely because of the multilevel terminal.

The Sigma terminal design uses four types of terminal windows on the display: flash, command/feedback, display, and view. The flash window is permanently allocated the top two lines of the display and is used to advise the user of the current date, time, system load average, and, more importantly, the arrival of new messages into the user's inbox. In addition, the flash window records the user's log-on clearance, the current open message file, and the date-time-group of the current open message. For the duration of the session, the flash window is classified at the user's log-on clearance.

The combined command/feedback window resides permanently within the next two lines of the display. The lower line is the command window, whereby the user enters commands and parameters to Sigma. The upper line is the feed-back window, whereby Sigma informs the user as to the status of command execution (e.g., command being processed, ambiguous or illegal command, security violation). As noted above, the command window is Unclassified most of the time, except when it is automatically upgraded by certain function keys to the level of the message file that is currently open.

The remaining lines below the command/feedback window may be used as a display window or as a view window, or the area may be shared by a display (upper half) and a view (lower half) window.

The display window is a user-editable window. Message-file surveys are displayed in the display window, and the user may add comments or keywords to selected entries. When the user issues a create-message command, Sigma displays a message form in the display window and the user fills in the desired fields. Existing messages can be displayed in the display window for editing or the addition of comments to certain fields.

The view window is a read-only window intended to provide a reference capability. The view window typically is used to display a message, text object, message-file summary, or a directory of message files or text objects, while performing some related operation within the display window (e.g., composing a message in the display window while referring to an existing message within the view window).

Both the display and view windows may be scrolled independently by moving the cursor into the particular window and depressing the desired scrolling keys.

The classification of a display or view window is governed by the classification of the message, message file, or text object that it contains. The classification can be determined by moving the cursor into the particular window and reading the security-level lights on the keyboard. Another set of security-level lights, situated adjacent to the display screen, reflects the highest classified window (omitting the flash window) on display.

4. MME SELECTION REQUIREMENTS

The MME-selection-requirement area of the security/privacy evaluation was solely an evaluation of the implementation of candidate message services and not of the security design. Criteria for this evaluation reflect the set of minimum security/privacy capabilities required for selection and operation at CINCPAC [3]. The criteria are grouped into nine categories, listed below along with the number of points each category contributes to the score in this area, with 100 points being available, and this area contributing 50% of the overall security/privacy score:

CATEGORY	POINT VALUE
Identification of security elements	15
Access controls	15
Log-on/role identification	5
Message filing/retrieval	10
Message distribution/annotation/keywords	10
Message composition	10
Downgrading	10
Coordination/release	15
Other - hardcopy, archiving, SSO facilities	10
	100

Most of this evaluation was performed by exercising each candidate service through scenarios of message-handling operations designed to demonstrate the presence or absence of the minimum security/privacy capabilities. The specific scenarios are not discussed here.

After a candidate service was used to perform the scenarios, the service was assigned a rating (1 to 5) in each of the nine categories. The rating was assigned according to the following guidelines:

GUIDELINE	RATING
The message service meets all the requirements with only a few minor deficiencies	5
The service has more than a few minor deficiencies	4
The service has numerous minor deficiencies or a major deficiency	3
The service has major deficiencies	2
The service meets none of the requirements	1

Because the point values in each of the nine categories are multiples of 5, it is simple to map ratings into points.

4.1 Identification of Security Elements (15 points)

Security-relevant message-service program modules shall be identified. General writedown capabilities (the capability to downgrade the security classification of a file, say, from Top Secret to Unclassified) shall be granted only to designated modules. Writedowns shall be monitored by the security system, and their security ramifications analyzed. Necessary software compromises required to meet specific MME requirements shall be documented. The need for trusted jobs (those software modules granted a writedown capability) and secure communications channels between them and the user shall be documented and either implemented or simulated.

BBN's Hermes

Hermes had no deficiencies and was assigned a rating of 5.

ISI's Sigma

Sigma had no deficiencies and was assigned a rating of 5.

4.2 Access Controls (15 points)

Nondiscretionary access control shall be used to enforce DOD clearance/classification security policy on messages, message fields, and message files. Recognized security classifications shall be Unclassified, Confidential, Secret, and Top Secret; information compartments are not supported.

Discretionary controls shall be used to implement privacy and need-to-know measures and should be applicable on both an individual and organizational basis. Discretionary controls should be used to restrict access to messages, message files, annotations/ comments on messages and message fields, as well as to other sharable objects, such as selectors/ filters for message retrieval and text objects.

BBN's Hermes

Although the access controls are adequate, there is a major deficiency and three minor deficiencies. The major deficiency is that the message classification field is simply a marking that could be edited by the user. Message classification is not enforced on access to the message.

A minor deficiency discovered is that, because of software bugs, information classification markings on the display screen can be changed by the user. Other minor deficiencies are that minimum access level on message files are not supported and there is no limit on the classification of information that can be stored in message files (no concept of message-file classification). Hermes was assigned a rating of 3.

ISI's Sigma

Access controls are adequate with one major and one minor deficiency detected. The major deficiency is that the discretionary access controls are inflexible. Discretionary access to message files, and the messages they contain, is enforced solely on a directory basis; thus, discretionary control is the same for all message files in the directory.

The minor deficiency is that minimum access level on message files is actually the message-file classification. For example, a Secret message file can be accessed only by users cleared to Secret and above (the essence of minimum access level), but the file may contain only information at Secret and below. Sigma was also assigned a rating of 3.

4.3 Log-on/Role Identification (5 points)

The message service must support security clearances on user names and terminals. The System Security Officer (SSO) will register users and their clearances on the service, assign passwords to users, and assign clearances to terminals.

A user logs on by entering a user name, submitting a password and, optionally, by specifying a log-on security level. The service should support a default log-on security level that can be set by the user. If not set by the user, the default should be the user's clearance level. The log-on security level must not exceed either the user's or the terminal's clearance. In effect, the log-on security level is the user's clearance for that session.

After logging on, users may then request to operate in an organizational role. The service shall authenticate that the user can assume the requested role. For each user, the SSO will register the roles that may be assumed. Clearances are not associated with organizational roles.

BBN's Hermes

Hermes is adequate, but the following minor deficiencies were noted. No settable default log-on security level is available. Also, specifying a log-on security level (session clearance) is cumbersome; the user is automatically logged on at his maximum level (his clearance); then he must set the session clearance to a lower level if desired. Hermes was assigned a rating of 5.

ISI's Sigma

Sigma is adequate, but the following minor deficiency was noted. No settable default log-on security level is available. Also, the following inconvenience (not a deficiency) was noted. Sigma authorizes role assumption through a password mechanism, and not by checking a list of legal users, thus making role assumption somewhat more awkward than necessary. Sigma was assigned a rating of 5.

4.4 Message Filing (10 points)

The message service must provide the user with message files for the grouping of messages. Although names of message files may be unclassified, it should be possible to assign a message-file classification so that only messages at or below that classification are included in the file. Also, for each message file the service should support a minimum access level. To access a message file, the user's log-on clearance must be equal to or exceed the file's minimum access level. The message service must also support discretionary access controls on files. Specific discretionary access rights (i.e., read, write, delete) to message files should be supported on both an individual and organizational basis.

The service should provide the user with tools (such as filters, selectors) to aid in message retrieval. If these tools are allowed to contain classified information, the message service must control access to them.

BBN's Hermes

Hermes is barely adequate in this category; two major deficiencies were detected. Filters and message sequences must be unclassified; hence, they cannot contain classified search criteria (subjects, keywords, references) - a serious deficiency. Also, when a user operating at a given level (say, Confidential) files or moves a message to another file, fields of the message that are classified above Confidential (and hence, invisible to the user) are also filed or moved into the destination message file. This should not occur, since the Confidential user must not be able to manipulate Secret or Top Secret information in any manner. Further, the results are potentially confusing to the user because he may not realize what caused the "invisible" transfer.

Two minor deficiencies were also detected. First, as noted in Section 4.2, Hermes supports neither minimum access level nor overall classifications on message files. Second, when a file is created, an internal message service writedown is performed before the user even confirms that a file is to be created. Hermes was assigned a rating of 2.

ISI's Sigma

Sigma also was barely adequate; two major deficiencies were detected. Sigma also has security problems with its file and move commands. The move command deletes file entries in the source message file before verifying the legality of the move. With the file command, Sigma informs the user that all specified entries have been filed in the destination file, even if some of the entries were not filed because of security (i.e., the classification of some of the specified messages exceed the classification of the destination message file).

Also, as noted earlier, discretionary access controls on message files are inflexible. Sigma was also assigned a rating of 2.

4.5 Message Distribution/Annotations/Keywords (10 points)

Message distribution takes the form of action assignments, cognizance (cog) assignments, and the forwarding of messages for information purposes. Entries to the Directorate-wide ACTION/COG LOG are made for each action and cog assignment automatically by the message service.

Annotations take several forms. Messages may be annotated during message distribution. Annotations can be made at any security level and discretionary access controls to the annotation, independent of discretionary access controls on the message itself, may be specified by the annotator.

Annotations may also be affixed to folder entries, both personal and organizational folders, as well as to readboards and the ACTION/COG LOG. Again, it should be possible to annotate at any level and to specify discretionary access rights to the annotation.

Keywords may be attached to messages at various times, e.g., during message distribution, message composition, and message filing. It should be possible to add keywords at either the unclassified level or at the level of the message text.

BBN's Hermes

Hermes contained several major deficiencies and was barely adequate. First, discretionary controls are not supported on annotations. Second, the identity of the annotator can be changed by the user. Third, the process of annotation, via the explode or comment command, is awkward and prone to error.

A minor deficiency is that the handling of the ACTION/COG LOG and annotations to it appeared to contain bugs. Hermes was assigned a rating of 2.

ISI's Sigma

Sigma was adequate, but some minor deficiencies were detected. Keywords cannot be attached directly to messages; rather they can only be associated with entries for messages in message files. A further restriction is that all keywords and annotations are made to message-file entries and are protected at the classification of the message file. Also, Sigma does not maintain a global ACTION/COG LOG that reflects all action/cog assignments and reassignments as messages were distributed throughout the CINCPAC organizational hierarchy. Sigma was assigned a rating of 4.

4.6 Message Composition (10 points)

Message composition may be initiated through either a reply command or a general composition command. In both cases the service must support an appropriate prompting mechanism. The reply command should automatically copy the subject and header of the message being answered into the subject and header of the message being composed. It should be possible to reply to a received message at a classification different from that of the received message. It should be possible to suspend the composition process to read other messages and retrieve references, then to resume composition without loss of state.

BBN's Hermes

Hermes is barely adequate in the message-composition category, containing several major deficiencies. The service does not prompt the user for message classification. The reply and forward commands are awkward, consistently placing the user at the wrong security level for message composition. Also, on reply, the To and From fields are copied down internally via a writedown without any apparent intervention by a trusted job. Further, correct information is not written into the Action and Info fields when a reply to a formal message is performed. Hermes was assigned a rating of 2.

ISI's Sigma

Sigma is adequate, with a minor descrepancy noted. The Reply Command does not copy the Subject field to the message in composition. Sigma was assigned a rating of 5.

4.7 Downgrading (10 points)

Users should be able to downgrade text objects and informal messages. Also, reclassification downward of message components during composition must be treated as a downgrade. The service must display the information to be downgraded to the user for confirmation. The service must display and downgrade no more than a screenful of information at a time, each screenful requiring explicit user confirmation. When composing a message, a user shall be prompted to declare a tentative classification. This shall be modifiable by the user at any time prior to message transmission. If the security classification of the message is thereby reduced, the operation shall be considered a downgrade.

BBN's Hermes

Hermes supports downgrading adequately. An inconvenience associated with downgrading a text object is that a copy is left at the higher classification and must be deleted by the user. Hermes was assigned a rating of 5.

ISI's Sigma

Sigma also is adequate, with a minor deficiency. It is cumbersome to downgrade a message in composition. The text of each message component or field must be individually excised as a text object and downgraded separately. Sigma was assigned a rating of 4.

4.8 Coordination/Release (15 points)

The message service must support both serial and parallel on-line coordination of formal message drafts. Differentiation shall be made between messages sent for information only and those sent for approval. Coordination specified for users not part of the experiment shall automatically generate a printout per coordinator for later manual distribution. Authorized release requests shall interrupt and override coordination at any point in the coordination process.

Messages released to the Local Digital Message Exchange (LDMX) shall be monitored by a confirm reprint. Release capability must be restricted to a group of users defined by the System Security Officer (SSO). Release may occur only from terminals granted release capability by the SSO.

BBN's Hermes

While Hermes performs adequately in the coordination/release category from a security/privacy standpoint, coordination and release are handled awkwardly. Thus, Hermes creates a situation that would induce security errors in an operational system. Therefore, Hermes was assigned a rating of 4.

ISI's Sigma

Sigma is adequate in this category and was assigned a rating of 5.

4.9 Other (10 points)

Included in the coordination/release category are three security/ privacy requirements published in the MME-selection-criteria document [3] that were relaxed for the evaluation. These are the requirements for hardcopy and archiving of messages and the SSO capabilities. Instead of implementing these requirements, each developer was required to submit a description detailing the means of supporting these requirements in the event of selection for CINCPAC.

BBN-Hermes

The BBN description is inadequate, suffering major deficiencies. Hardcopy, archiving, and most of the required SSO capabilities are not addressed at all. The SSO capabilities that are addressed are simply those necessary to enable the Hermes test system to run (e.g., registration of users, terminals, and passwords using the TENEX WHEEL capability). Hermes was assigned a rating of 2.

ISI's Sigma

The ISI description is more thorough, with the exception that some of the SSO functions are not discussed fully. It is considered adequate. Sigma was assigned a rating of 4.

4.10 Scoring Summary

The table below reflects the scoring for the MME-selection-requirement area of evaluation.

CATEGORY	POINT VALUE	BBN	ISI
Identification of security elements	15	15	15
Access controls	15	9	9
Logon/role identification	5	5	5
Message filing/retrieval	10	4	4
Message distribution/ annotation/keywords	10	4	8
Message composition	10	4	10
Downgrading	10	10	8
Coordination/release	15	12	15
Other - hardcopy, archiving, SSO facilities	10	4	8
	100	67	82

5. SECURE SYSTEM STRUCTURE

In the category of secure system structure, the designs submitted by the developers were evaluated as possible bases for a secure message service for a multilevel user environment. The designs were evaluated by considering each design's demands on a security kernel in four functional categories: secure file system, secure process structure, secure multilevel terminal and terminal multiplexer, and secure process coordination. In each category, the operating system must provide primitive functions that ensure the secure flow of information within the message service.

The primary consideration was the following. Given present security kernel technology and a perception of trends in hardware and software development that could relate to kernel technology in the immediate future, can a kernel be built and verified to secure an implementation of the design? The fundamental criterion up which this determination was based was the size and complexity of the required kernel. Size has a direct bearing on the verifiability of a kernel, and complexity can determine the feasibility of even building a kernel.

Again, ratings from 1 to 5 were assigned, according to the following guidelines:

GUIDELINE	RATING
The design is compatible with present kernel technology and state-of-the-art computer hardware; it is clear that a kernel can be built and verified to support the design's requirements. Or, alternatively, the design demands a larger and/or more sophisticated kernel, but significant improvements to the user and/or system performance are documented to justify the increase in kernel size and/or complexity.	5
The design makes demands that clearly cannot be satisfied by current or predicted kernel technology, and the documented improvements gained by this design clearly do not justify the additional requirements.	1

Ratings of 2, 3, and 4 are assigned for designs between the two extremes.

This evaluation contributed 40% of the score on security/privacy, and there were five categories worth a total of 100 points. Seventy-five points were assigned to evaluating design requirements in the four functional kernel categories. The remaining 25 points were awarded on the basis of the compatibility of the design with the user interface presented by the developer's candidate message service.

The five categories of evaluation and their respective point values are listed below:

CATEGORY	POINT VALUE
Secure file system	20
Secure process structure	15
Secure multilevel terminal	20
Secure process coordination	20
Compatibility of user interface	25
	100

5.1 Secure File System (20 points)

A sophisticated message service must manipulate several types of objects which are visible to the user: message files, text objects, message selectors for search and retrieval, and templates for printing formats. Most of these objects may contain classified information; thus, the message service must control access to them. The secure message service developer must implement these objects in terms of the security kernel's basic protection object or, more specifically, in terms of the secure file system.

Here the evaluation consisted of investigating the type of file system appropriate to protecting the design's secure-message-service objects and determining the feasibility of building the file system.

BBN Design

The BBN design protects individual message fields. Messages are stored in message files that, transparent to the user, are implemented as four physical files internally, one at each of the four security levels. Each internal file contains only those fields of the message file that are classified at the level of the internal file. This arrangement can be efficiently and easily supported by a secure flat-file system with minimum file sizes on the order of a memory page (e.g., 512 or 1024 words). Text objects and other message service tools can also be supported by the same type of file system. The demands of the BBN design on the file system were consistent with current kernel technology. BBN was assigned a rating of 5.

ISI Design

The ISI design protects messages as a whole. All fields of a message are protected at the level of the message classification. Because all users share a common message data base, message files are simply collections of message "citations." A Secret message file is implemented as a Secret, Confidential, and Unclassified internal file. A message citation is stored in the internal file at the level of the message being cited. The Secret internal file also includes the message file structure (the number of citations, or entries, in the file) and any comments or keywords on file entries. Text objects and message selectors are also implemented as internal files at the level of the object or selector. The ISI design documentation states that the file system must support many small files (about 1000 bytes) as well as many large files (about 10,000 bytes). This requirement is consistent with current kernel technology. ISI was assigned a rating of 5.

5.2 Secure Process Structure (15 points)

To permit the message-service user to read and write messages of different classifications in one session, message-service processes must exist at all classification levels. This means that several processes at various security levels must operate on behalf of each user. The demands of a secure message service on the process support capabilities (process activation, deactivation, interprocess communication) of a kernel are an important consideration and a factor that may limit the number of users that the service can support.

BBN Design

The BBN design uses a complete message-service process at each security level up to the user's clearance. However, only one process need be active at a time. During a session, the user switches operating level by command to read or write message fields at that level and below. By so doing, the user activates a process at the desired level and deactivates a process at the previous level. Process activation occurs serially with a minimum number of active processes systemwide.

One concern with the BBN design is that because a complete message service must be deactivated and another complete message service activated for each change in security level, the changes in security level in an operational environment could produce a high system load. Nonetheless, the process requirements of the BBN design can be satisfied by present kernel technology. BBN was assigned a rating of 4.

ISI Design

The ISI design requires many processes, possibly as many as 25 per user. Although not all of these processes must be active at the same time, two to four processes must be activated at each of the user's valid security levels in order to execute a command.

There is a concern that the demands of the ISI design for quick and efficient process switching and interprocess communication may seriously limit the number of users that could be serviced with reasonably good response. ISI was assigned a rating of 3.

5.3 Secure Multilevel Terminal (20 points)

To work efficiently, the user of a secure message service must be able to concurrently read and write information at different security levels. For example, the user may wish to view a Confidential message while composing or editing a message at Top Secret. The concept of a multilevel visual-display terminal addresses this requirement, but it also causes additional security concerns.

First, a means of securely transferring multilevel information between the multilevel terminal and message service processes is essential. The security kernel must provide primitives that support these secure transmission paths.

Second, because information at multiple classification levels will be stored within the terminal, some assurance is needed that terminal-resident information is not compromised. The terminal microcomputer program must be verified to perform its functions correctly.

Third, since multilevel information is being processed by the terminal, effective mechanisms must be used to maintain user cognizance of the classification of information being read or written, in order to minimize the risk of accidental information compromise by the user.

BBN Design

The BBN design makes use of a multilevel terminal that is teletype oriented. There is no concept of terminal windows (these are essentially information containers or files). Information input from the keyboard is transmitted directly to the host by the terminal program, character by character, and is not stored within the terminal. Only information generated by the host (echoed input, message service output) is stored in terminal memory and displayed. The design includes no local terminal functions.

The BBN terminal offers no major security constraints to the development of a secure message service, provided the terminal program is verified to make the terminal act like a teletype. That is, all information input from the keyboard must be transmitted directly to the host unaltered by the terminal. Information received from the host must be stored and displayed unaltered.

Kernel primitives necessary to transfer information securely between the windows of the BBN terminal and the message service processes present no significant problems, because there are no classified windows to be concerned with. However, the kernel must be able to easily detect user requests for operating level changes so that terminal input can be switched to the appropriate process. This detection would be simpler if the terminal design used security level function keys "wired" to the kernel, instead of the command strings Unclassified, Confidential, Secret, Top Secret. On output to the terminal there is no need for maintaining the separation of information at different security levels, since there are no windows. The kernel must provide security classification markings along with the output.

The user/terminal interface design adequately maintains user awareness of current operating level and the classification of all information on display. A condition was detected, however, whereby the user could alter the classification markings affixed to each line of the display. These markings must be inviolable.

Although the basic teletype approach is relatively simple to secure (versus a multilevel terminal with windows), it provides none of the advantages afforded by the multilevel terminal. The user cannot scroll and hence cannot refer to the full contents of messages and message file surveys while performing other functions. Clearly, the user of the BBN system would be better served by a hardcopy terminal. BBN was assigned a rating of 4.

ISI Design

The ISI design treats the terminal as a true multilevel terminal, whereby terminal memory is divided into a small number of windows of independent classifications. These windows are used by the message service under the auspices of the security kernel. For each terminal in use, the kernel maintains a table that contains the classifications of all active windows. Information received from the terminal is marked with window of origin and the kernel switches the information to a message service process at the level of the window. The kernel also ensures that terminal-bound information is destined for a window whose classification is equal to or greater than the message-service process of origin.

Any increases in kernel size and complexity due to the incorporation of kernel primitives for multiplexing and controlling the multilevel terminal would be manageable and would not seriously impact either construction or verification of the kernel. Considering the significant advantages that the multilevel terminal offers to the user, any attendant increases in kernel size and complexity are acceptable. Some additional kernel capabilities required to support the multilevel terminal are noted. The kernel must understand the protocol of communicating (NOTICEs and DISPATCHes) with the terminal microcomputer. The kernel must be able to determine the window of destination in DISPATCHes generated by the message service and the window of origin of NOTICEs generated by the terminal program. Further, to maintain the active window classification table, the kernel must recognize all DISPATCHes involved in window allocation and deallocation.

Also, in an eventual implementation of the ISI design for the multilevel user environment, the terminal microcomputer program that implements the correct operation of the multilevel terminal must be verified. Specifically, integrity of the terminal windows must be validated; information must not leak between windows or be otherwise compromised within the terminal. Verification of the multilevel terminal program is attainable, given a comprehensive test and evaluation strategy.

Finally, the ISI interface design keeps the user well informed of any information being read or written. By moving the cursor into a given window, the keyboard security-level lights can be used to determine the classification of information contained within the window, while security-level lights on the display reflect the highest classified window on display. ISI was assigned a rating of 5.

5.4 Secure Process Coordination (20 points)

With a number of message-service processes operating at different security levels on behalf of each user, interprocess communication is necessary to coordinate internal activities. There are occasions when processes at higher security levels must transmit information (such as message identifiers, error conditions) to processes at lower security levels. These writedowns must be controlled, since there can be no assurance that higher level processes are merely coordinating internal activity and not actually compromising classified information. The security kernel must support a form of controlled, indirect communication between message service processes at higher and lower security levels. Use of these writedown paths must be minimized and tied to explicit user action, such as depression of a function key.

BBN Design

Users of a secure message service implemented from the BBN design would tend to operate at their maximum level. Commands are entered at the top operating level, and the kernel writedown primitives must be used to coordinate execution with lower level processes. Further, quite a bit of information of several types must be sent down: command identifiers, lists of message numbers, and file and message service object names up to 39 and 50 characters long, respectively. Also, in some instances (commands without arguments) information is written down without explicit user confirmation action.

These design requirements raised serious questions as to whether a kernel could sufficiently monitor and control the use of the writedown primitives. BBN was assigned a rating of 3.

ISI Design

The ISI design maintains a predominantly unclassified command window. Most commands and function keys are interpreted (with parameters) by an unclassified command processor. For unclassified commands, the kernel must support a one-bit writedown path by which higher level processes can signal the presence of an error condition to the unclassified command processor and execution state. The kernel will not permit the writedown unless a function key has been depressed (either the command-function key or execute key). The FILE, MOVE, and DELETE file-entry commands require the use of larger writedown paths to pass down lists of file-entry numbers. The kernel sorts the entry list, informs the user of the number of entries being filed, moved, or deleted, and requires the user to acknowledge the event by depressing the particular command's function key.

The ISI design minimizes the number and size of writedown paths required to coordinate internal activities within a secure message service. ISI was assigned a rating of 5.

5.5 Compatibility of User Interface (25 points)

The secure user-interface presented by each candidate message service must be consistent with the developer's design for a secure message service. In this way, the selected developer's design for a secure service could be implemented during a later stage of the MME with minimum perturbations at the user interface and without extensive retraining.

BBN Design

No incompatibilities were found between the BBN candidate service's secure user interface and their design for a secure service. BBN was assigned a rating of 5.

ISI Design

No incompatibilities were found between the ISI candidate service's secure user interface and their design for a secure service. ISI was assigned a rating of 5.

5.6 Scoring Summary

The table below reflects the scoring for the secure-system-structure area of evaluation.

CATEGORY	POINT VALUE	BBN	ISI
Secure file system	20	20	20
Secure process structure	15	12	9
Secure multilevel terminal	20	16	20
Secure process coordination	20	12	20
Compatibility of user interface	25	25	25
	100	85	94

6. CERTIFICATION

In the final area of the security/privacy evaluation, the certifiability of a system implementation of each developer's design for a secure service was examined. This area contributed 10% of the overall score on security/privacy.

Certification is essentially a policy issue. Whereas technical criteria can be defined that verify characteristics for hardware and software, certification addresses the more fundamental issue of what level of technical countermeasures are needed to reduce the risks of compromise to an acceptable level. A design is certified, then, on the basis of perceived risks.

Within a computer system, potential threats of information compromise come from all individuals who access the system, such as users, operators, system programmers, maintenance personnel, and others normally permitted access, as well as agents, saboteurs, or others who may attempt to gain access illegally. The certifiability of each design was judged in a series of four user environments, such that increasingly fewer controls are exercised on the accessors and the chances increase that untrustworthy and maliciously inclined individuals may be granted access to the system. The four environments and their respective point values are listed below:

Strictly controlled	10 points
Controlled	40 points
Semicontrolled	40 points
Open	10 points

Again, ratings of 1 to 5 were assigned in each enviroment according to the following guidelines.

GUIDELINE	RATING
A design with an excellent chance for certification	5
A design that is adequate for certification although some doubt remains	4
A design with a questionable chance for certification because of several deficiencies in the design	3
A design with a doubtful chance for certification because of major deficiencies in the design	2
A design clearly not certifiable because of excessive risks.	1

6.1 Strictly Controlled Environment (10 points)

A strictly controlled environment is defined to be one in which all personnel who gain access to the system are cleared to a single level. Typically, the computer system may be accessed only from within an area that is well defined by physical, personnel, and administrative controls. Such controls are generally recognized as effective in denying system access to individuals not cleared to a particular security level.

Multilevel message systems dedicated to users cleared to a particular level are essentially single-level systems. Whereas internal controls should be used to protect against accidental leakage of information across security levels, in essence all information within the system is protected at the dedicated level by the external controls.

The only area requiring extensive hardware and software controls in this environment is the releasing, or transmission, of messages to an environment at a level lower than the dedicated level. Controls are necessary here to ensure that the classification markings on a message adequately correspond to the information contained within that message.

BBN and ISI Designs

We concluded that message services implemented from both the BBN and ISI designs would have excellent chances for certification in this environment. Since untrustworthy individuals are denied access by external controls, the remaining perceived risks lie with the release of messages to the external environment. Both BBN and ISI implementations could be certified, provided the release of messages is controlled by additional hardware and/or software mechanisms. Both systems were assigned a rating of 5.

6.2 Controlled Environment (40 points)

A controlled environment is defined to be one in which all individuals who may gain access to the system have similar clearances, and access to the system is again controlled by external physical, personnel, and administrative controls. This environment differs from the strictly controlled environment in that users of more than one clearance (e.g., Secret and Top Secret) may have concurrent access to the system.

The same security measures necessary for certifying single-level systems must be used here, but now the internal controls need to be strong enough to conteract the perceived threat of penetration. As a minimum, the control must be strong enough to protect against both accidental disclosure or compromise and the over zealous user attempting to bypass the controls. In this environment, the threat of penetration from the malicious user or the software developer's trap door is not as great as in the semicontrolled or open environments.

BBN and ISI Designs

Experience with the Air Force Data Services Center MULTICS system [18,19], which operates in a similar environment, shows that implementations of both the BBN and ISI designs have excellent chances at certification in the controlled environment. Both BBN and ISI were assigned a rating of 5.

6.3 Semicontrolled Environment (40 points)

The semicontrolled environment is the general case for secure message systems and is the environment of primary interest in the certification evaluation. In this environment, some form of physical, personnel, and administrative controls are exercised on all users. However, users of a wide range of clearances (e.g., Confidential through Top Secret, with compartments) may concurrently access the system. Therefore, there is a much greater risk that potentially malicious users may have access to a system processing highly sensitive information.

Little can be said about policy governing the necessary safeguards to certify a system for this environment. Although all the external security measures mentioned above are still necessary to secure the system, the internal hardware and software controls are the final line of defense; therefore they must address the potential for abuse of the system. It would seem that the most advanced verification technology would be required for validation of a system in this environment.

BBN Design

Several features of the BBN message system are troublesome in this environment. Chief among these is the requirement that the user must read his messages at his highest level to ensure that he is seeing all of the message fields. However, these disadvantages should not be sufficient to deny certification. Therefore, BBN was assigned a rating of 4.

ISI Design

The ISI design should have an excellent chance of being certified in this environment. Because the command parser operates at the unclassified level, it is convenient for users to work at the lowest possible level; this is a significant advantage. ISI was assigned a rating of 5.

6.4 Open Environment (10 points)

A fully open environment is defined to be one in which personnel with a wide range of clearances, including personnel who have no clearance, have concurrent access to the system and in which little, if any, external controls are placed on access by the uncleared users to the system. An example of this type of environment would be a system that allowed concurrent access by Top Secret users and uncleared users using dial-up phone lines. Because no such system exists today, even less can be said about the certification of such a system than was said about systems in the semicontrolled environment. It would seem that nothing but the most advanced verification technology would suffice for certification of a system in an open environment. In addition, threats such as traffic analysis and covert communication paths would almost assuredly have to be addressed for systems operating in the open environment.

BBN Design

The large number of writedowns, some of which are not user confirmed, coupled with the problems mentioned in the evaluation of the BBN system in the more controlled environments makes certification in this environment questionable. Therefore, a rating of 3 was assigned.

ISI Design

The ISI design employs as few writedowns as possible to achieve a desirable user interface. These writedowns all require some form of user interaction. However, the fact that writedowns are occurring raises some minor doubt as to the possible certification. Therefore ISI was assigned a rating of 4.

6.5 Scoring Summary

The table below reflects the scoring summary for the certification area of evaluation:

CATEGORY	VALUE	BBN	ISI
Strictly controlled	10	10	10
Controlled	40	40	40
Semicontrolled	40	32	40
Open	10	6	8
	100	88	98

7. CONCLUSION

The Security/Privacy Evaluation Subcommittee recommends that ISI's Sigma message service be selected for use in the Military Message Experiment at CINCPAC. As evident from the scoring, three major considerations form the basis for this conclusion: the user interface, the secure design, and the appropriateness for future message systems.

Several factors have contributed to the pleasing secure user interface presented by Sigma. The integration of the multilevel terminal features into the Sigma message service makes the user fully conscious of the classification of the information being read and written -- but unobtrusively. In most instances, Sigma places or guides the user into the appropriate security level.

Another factor evident from the evaluation is that the Sigma develop ment team has a clear understanding of the CINCPAC requirements and operations; thus, Sigma is tailored closely to the CINCPAC needs.

Because Sigma is driven predominantly from the Unclassified level, it has only a minimal reliance on internal writedowns for coordination between the levels. It is reasonably clear that a kernel could be built and verified to support the ISI design.

It is the consensus of the subcommittee that Sigma represents the first step in the development of future secure message-processing systems. By combining a pleasing and effective secure interface with a sound and securable underlying system structure, Sigma demonstrates that a message service can be both secure and easy to use.

8. REFERENCES

1 Memorandum of Agreement between Commander, Naval Telecommunications Command; Commander, Naval Electronic Systems Command; Director, Defense Advanced Research Project: Agency; and Commander in Chief, Pacific; dated 1 Dec. 1975.

2 N.C. Goodwin, J. Mitchell, and P.S. Tasker, "Concept of Operations for Message Handling at CINCPAC," MTR-3323, Oct. 1976.

3 E.H. Bersoff and S.H. Wilson, "Requirements for a Secure Military Message Processing System," NRL Memorandum Report, in preparation.

4 N.C. Goodwin, J. Mitchell, and S.W. Slesinger, "Test Plan for Military Message Handling Experiment," MTR-3268, Vol. I, The Mitre Corp., July 1976.

5 N.C. Goodwin and S.W. Slesinger, "Test Procedures for Military Message Handling Experiment," in preparation, The Mitre Corp., Oct. 1976.

6 T.A. Linden, "Operating System Structures to Support Security and Reliable Software," ACM Computing Surveys 8, (No. 4), 409-445 (Dec. 1976).

7 D.E. Bell and L.J. LaPadula, "Secure Computer Systems Mathematical Foundations and Model," MTR-244, The Mitre Corp., Oct. 1974.

8 K.G. Walter, W. F. Ogden, W.C. Rounds, F.T. Bradshaw, S.R. Ames, Jr., and D.G. Shumway, "Primitive Models for Computer Security," ESD-TR-74-117, Case Western Reserve University, Jan. 1974.

9 C.J. Popek and C.S. Kline, "Verifiable Secure Operating System Software," Proc. 1974 AFIPS National Computer Conference, Vol. 43, pp. 145-151, AFIPS Press, Montvale, New Jersey, 1974.

10 D.E. Bell and L.J. LaPadula, "Secure Computer Systems," ESD-TR-73-278, Vol. I-III, The Mitre Corp., Nov. 1973 - June 1974.

11 R.M. Graham, "Protection in an Information Processing Utility," Communications of the ACM 11 (No. 5), 365-369 (May 1968).

12 D.E. Bell and E.L. Burke, "A Software Validation Technique for Certification, Part I: The Methodology," ESD-TR-75-54, Vol. I, The Mitre Corp., April 1975 (AD 009849).

13 S.R. Ames, "File Attributes and Their Relationship to Computer Security," ESD-TR-74-191, Masters' Thesis, Case Western Reserve University, June 1974 (AD A002159).

14 J.K. Millen, "Security Kernel Validation in Practice," Communications of the ACM 19 (No. 5), 243-250 (May 1976).

15 L. Robinson and K.N. Levitt, "Proof Techniques for Hierarchically Structured Programs," Computer Science Group, Stanford Research Institute, Jan. 1975.

16 L. Robinson, P.G. Neumann, K.N. Levitt, and A.R. Saxena, "On Attaining Reliable Software for a Secure Operating System," pp. 267-284 in 1975 International Conference on Reliable Software, Los Angeles, California, Apr. 1975.

17 J.D. Tangney, S.R. Ames, Jr., and E.L. Burke, "Security Evaluation Criteria for MME Message Service Selection," MTR-3433, The Mitre Corp., June 1977.

18 "Design for Multics Security Enhancements," Honeywell Information Systems, ESD-TR-74-176, Electronic Systems Division (AFSC), Hanscom Field, Dec. 1973.

19 S.R. Ames, "A Security Compliance Study of the Air Force Data Services Center Multics System," MTR-3065, The Mitre Corp., June 1975.

APPENDIX A

SUMMARY OF SCORING RESULTS

	Point Value	BBN	ISI
MME Selection Criteria - 50%			
Identification of security elements	15	15	15
Access controls	15	9	9
Log-on/role identification	5	5	5
Message filing	10	4	4
Message distribution/annotations/ keywords	10	4	8
Message composition	10	4	10
downgrading	10	10	8
Coordination/release	15	12	15
Other - hardcopy, archiving, and SSO facilities	10	4	8
	100	67	82
Secure System Structure - 40%			
Secure file system	20	20	20
Secure process structure	15	12	9
Secure multilevel terminal/ multiplexer	20	16	20
Secure process coordination	20	12	20
Compatibility of user interface	25	25	25
	100	85	94
Certifiability - 10%			
Strictly controlled	10	10	10
Controlled	40	40	40
Semicontrolled	40	32	40
Open	10	6	8
	100	88	98

Overall Score

BBN Design

MME requirements	67	x	0.50	=	33.5
Secure system structure	85	x	0.40	=	34.0
Certification	88	x	0.10	=	8.8
					76.3